

OpenFPGA: Automate the Prototyping of Customizable FPGAs and CAD tools

Prof. Pierre-Emmanuel Gaillardon, Dr. Xifan Tang
Aurelien Alacchi, Baudouin Chauviere, Edouard Giacomin

University of Utah

IDEA and POSH Phase I Integration Session and Demos
Detroit, MI

15 July 2019 – 19 July 2019





Open-Source FPGA IP Generator

- FPGAs' ever-increasing role in modern computing systems



Wired and wireless communications



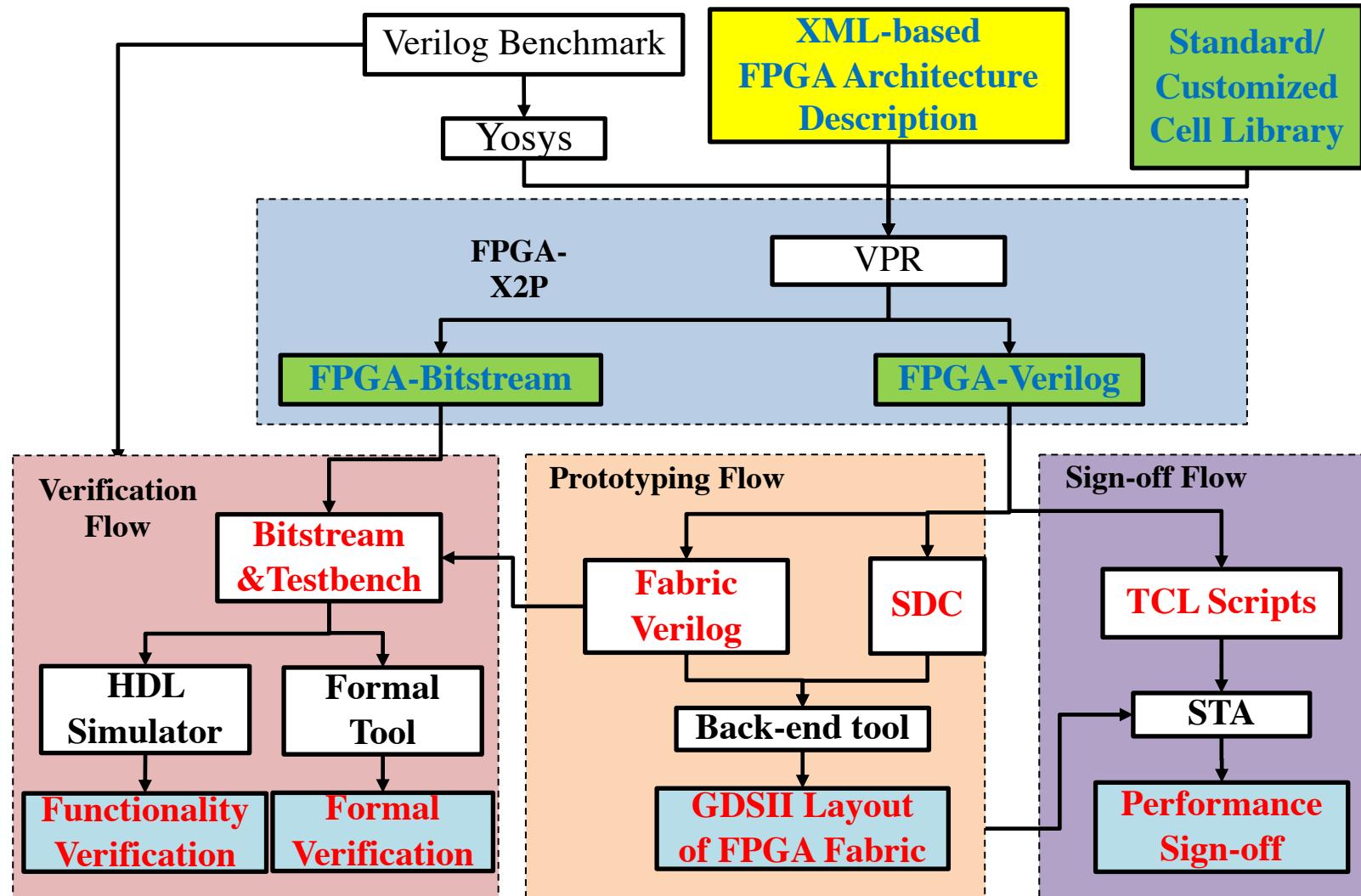
Audio and video broadcasting



Data center

- Prototyping FPGAs is traditionally a cumbersome process
 - Considerable manual layout (Groups of hardware engineers)
 - Ad-hoc design tool development (Groups of software engineers)
 - **Year-long development cycles**
- Open-FPGA: The First open-source FPGA IP generator
 - Enable a rapid prototyping flow for FPGA IPs
 - Customizable FPGA architecture and instant bitstream support
 - Enriched verification/design techniques for fabric and implementation

Automate FPGA development using a semi-custom design approach





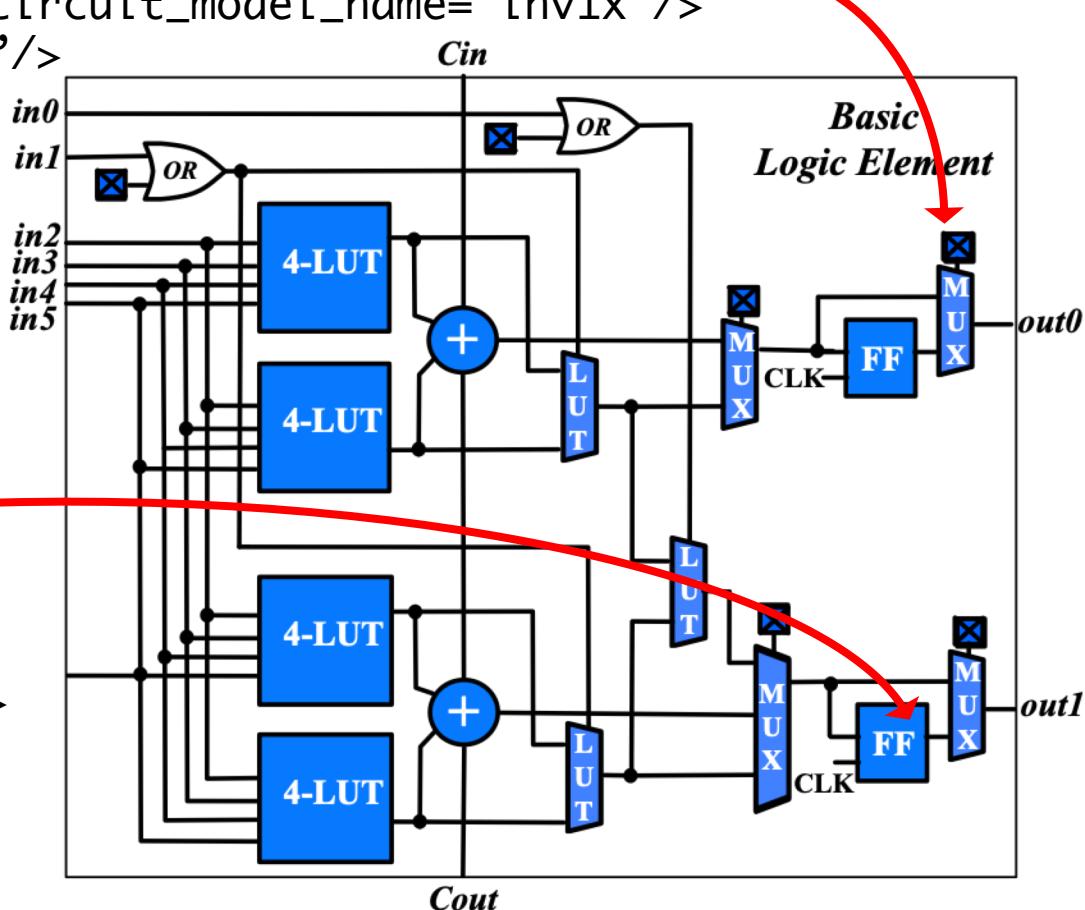
Architecture Description Language

XML definition for a MUX to be auto-generated in Verilog

```
<circuit_model type="mux" name="mux_1lvl">
  <design_technology type="cmos" structure="one-level"/>
  <input_buffer exist="on" circuit_model_name="inv1x"/>
  <output_buffer exist="off"/>
  ...
</circuit_model>
```

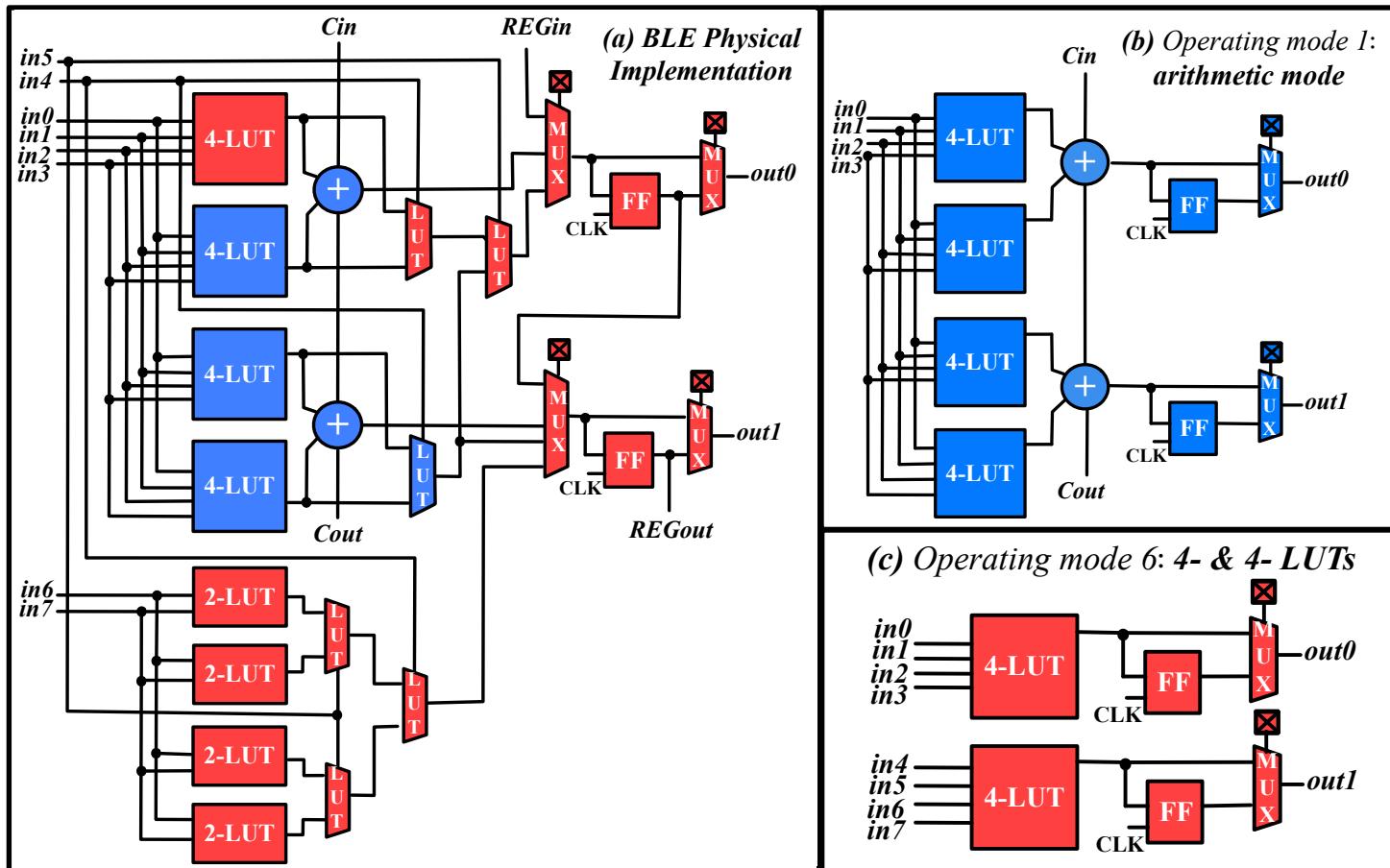
XML definition for a FF to be mapped to standard or customized cell

```
<circuit_model type="ff"
  name="dff"
  verilog_netlist="ff.v">
  <port name="D" size="1"/>
  <port name="CLK" size="1"/>
  <port name="Q" size="1"/>
  ...
</circuit_model>
```



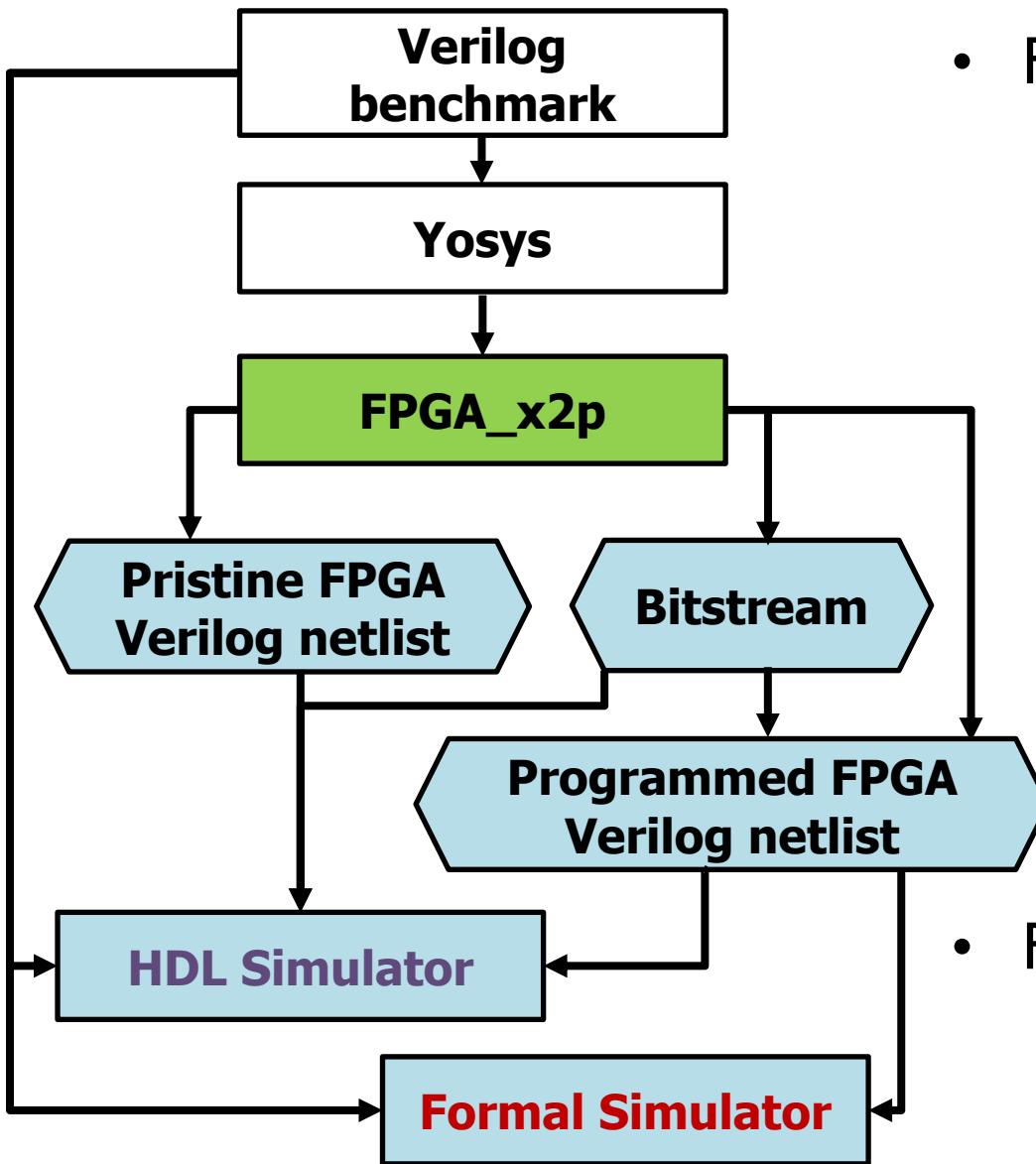
Generic Support on Multi-mode CLB

- Separated XML description for different modes of a CLB
 - An [example](#) of 6-mode BLE with adders, fracturable LUTs etc.
- Verilog and Bitstream will be automatically generated based on the physical implementation





Verification Techniques



- Functional Verification
 - Programmed FPGA:
 - > Benchmark Functionality only
 - Pristine FPGA:
 - > Loading bitstream
 - > Functionality
 - Commercial and open source simulator
 - > Mentor Modelsim
 - > Icarus Verilog
- Formal Verification
 - Commercial simulator:
 - > Synopsys Formality



Verified benchmark

- 40 Benchmarks went through OpenFPGA flow and has been verified
- Formal verification provides 100% fault coverage on implementation

Benchmarks	NAND Gate Equivalent	Verification runtime
PicoRV ^[1]	72,460	1 hour
Smithwaterman ^[2]	542,650	2.5 hours
EPFL suite	81,370	< 1 hour
MCNC big20 suite	20,950	< 1 hour

[1]: January 2019 best achievement -> PicoRV

[2]: July 2019 best achievement -> Smithwaterman

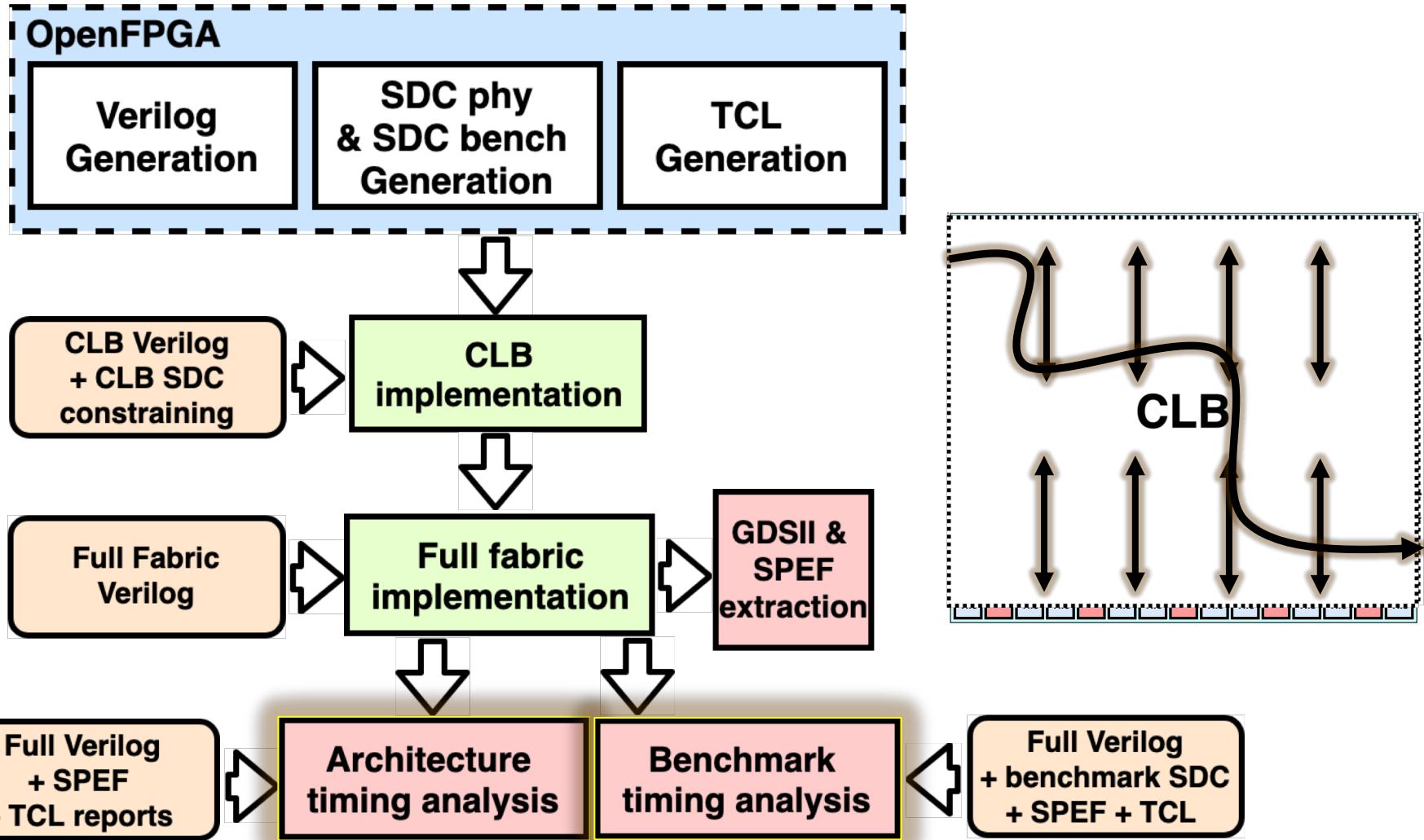
$$[2] = [1] \times 7.5$$



Live Demo STA

CLB Signoff:
Critical Path Analysis

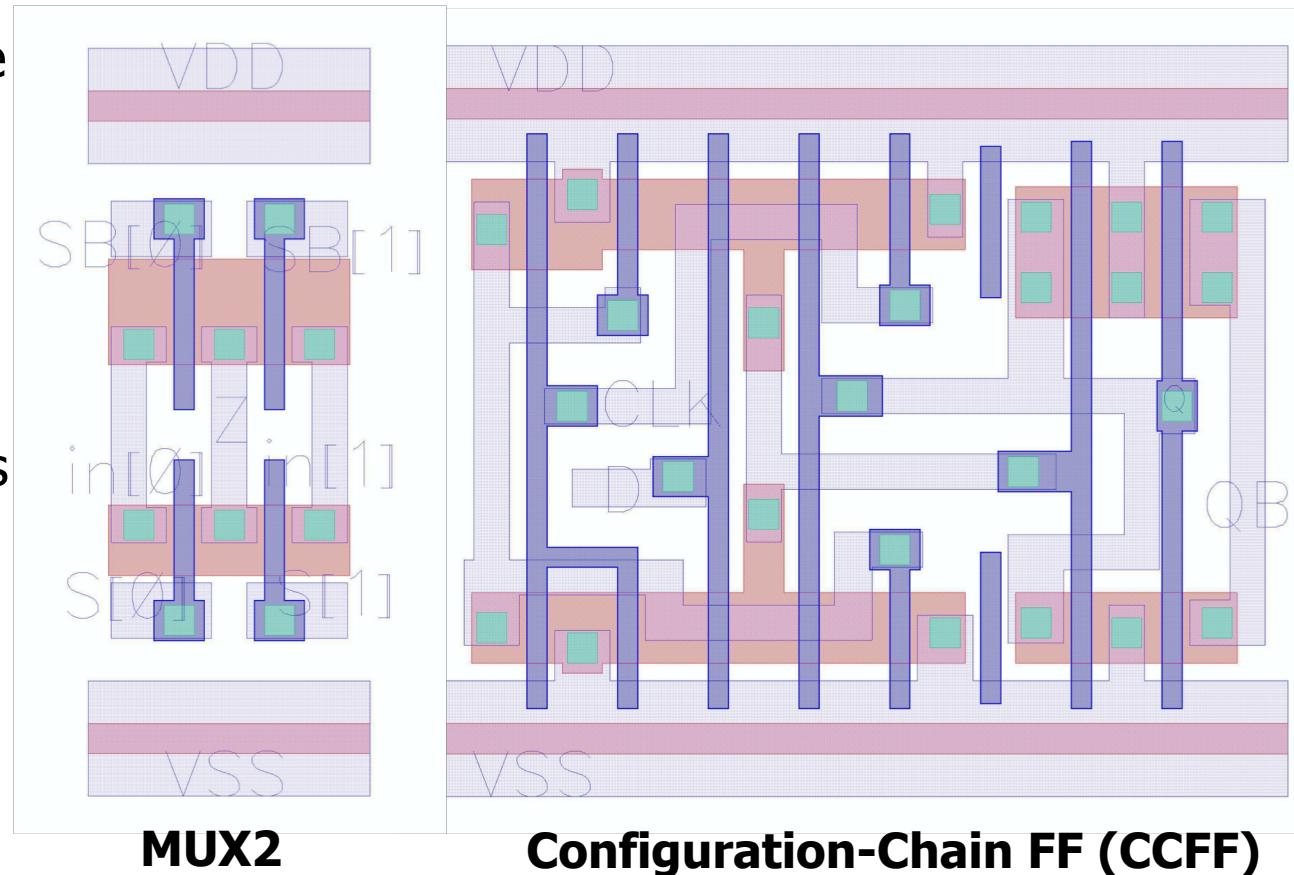
Post OpenFPGA working flow (P&R + STA)



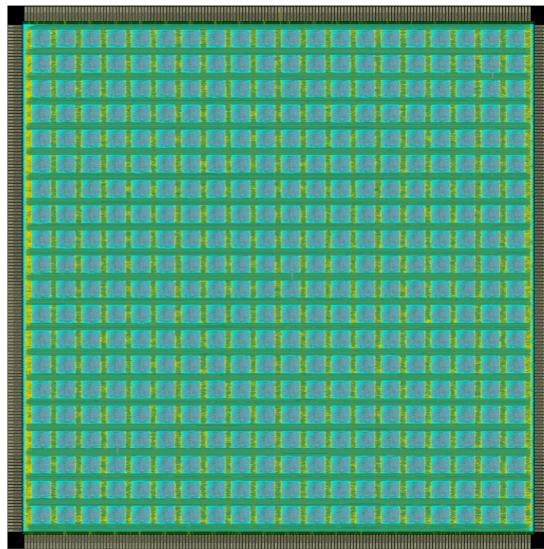
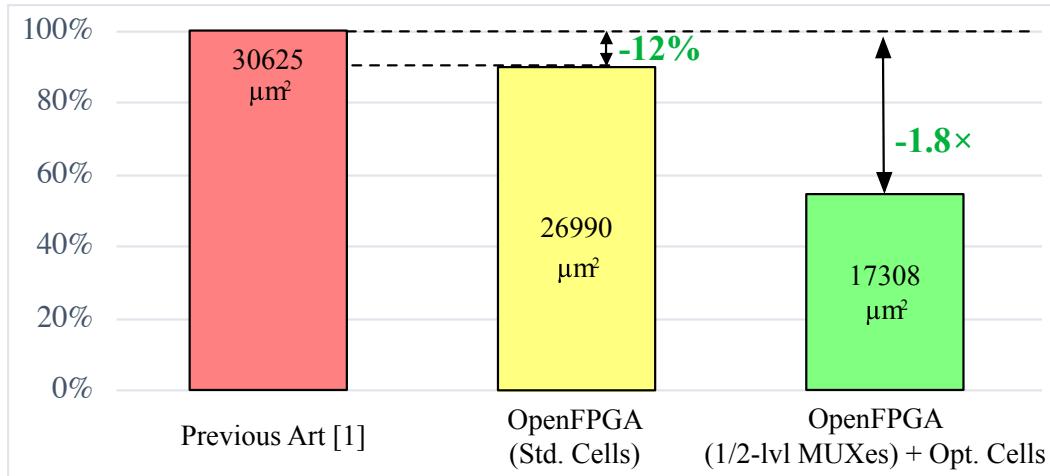


Performance Analysis

- Cell library has a strong impact on the FPGA metrics: 90% of the FPGA is made of CCFF and multiplexers
- Tgate-based MUX2 is 2.5x smaller than from the SC library
- CCFF is 1.8x smaller than a regular DFF



[1] B. Grady et al., Synthesizable Heterogeneous FPGA Fabrics, IEEE International Conference on FPT, 2018, pp. 1-8.

20x20 FPGA

- Post-layout evaluation using a commercial 40nm node and a Stratix IV-like architecture
- Using an optimized cell library (multiplexers, CCFF):
 - 1.8x area reduction**
 - 3x delay reduction**

Path Type Delay (ns)	Previous work [1]	OpenFPGA
5-LUT	0.46	0.14
6-LUT	0.5	0.15
1-bit Adder	0.7	0.54
20-bit Adder	1.63	1.10
Local Routing	0.27	0.12
L-4 track	2.53	0.40
L-16 track	4.02	0.78
Average	1.44	0.46

[1] B. Grady et al., Synthesizable Heterogeneous FPGA Fabrics, IEEE International Conference on FPT, 2018, pp. 1-8.

- **Fully functional XML to Prototype flow (FPGA-X2P) supporting homogenous multi-mode FPGA fabrics**
 - XML-to-Verilog generator
 - XML syntax to customize multi-mode CLBs
 - Tested up to 640k-gate FPGA
 - Verilog-to-Bitstream generator
 - Verilog testbenches for functionality/formal validation
 - >40 tested benchmarks up to 540k-gate FPGA implementations
- **Automatic backend flow for homogenous FPGA**
 - 20×20 FPGA layout using a commercial 40nm node
 - 2× area and 3× performance improvement over previous arts
- **OpenFPGA alpha release with tutorials**
 - Github repository: <https://github.com/LNIS-Projects/OpenFPGA>
 - Online documentation: <https://openfpga.readthedocs.io/en/master/>





Thank you! Q & A

The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.